

METHOD AND APPARATUS FOR INITIATING STRONG ENCRYPTION USING EXISTING SSL CONNECTION FOR SECURE KEY EXCHANGE

By:

CHRISTOPHER J. FRANTZ
ANDREW C. CARTES
E. DAVID NEUFELD
JOHN R. CARPENTER
ANDREW BROWN

EXPRESS MAIL MAILING LABEL	
NUMBER:	EV-017 056 967 US
DATE OF DEPOSIT:	January 4, 2002
<i>Pursuant to 37 C.F.R. § 1.10, I hereby certify that I am personally depositing this paper or fee with the U.S. Postal Service, Express Mail Post Office to Addressee service on the date indicated above in a sealed envelope (a) having the above-numbered Express Mail label and sufficient postage affixed, and (b) addressed to the Assistant Commissioner for Patents, Washington, D.C. 20231.</i>	
January 4, 2002 Date	<i>Carla Deblaw</i> Carla Deblaw

METHOD AND APPARATUS FOR INITIATING STRONG ENCRYPTION USING EXISTING SSL CONNECTION FOR SECURE KEY EXCHANGE

5

BACKGROUND OF THE INVENTION

1. Field Of The Invention

This invention relates generally to remote server management in networked computer systems and, more particularly, to providing secure communications paths for remote server management communications.

2. Background Of The Related Art

This section is intended to introduce the reader to various aspects of art which may be related to various aspects of the present invention which are described and/or claimed below. This discussion is believed to be helpful in providing the reader with background information to facilitate a better understanding of the various aspects of the present invention. Accordingly, it should be understood that these statements are to be read in this light, and not as admissions of prior art.

20

Since the introduction of the first personal computer (“PC”) over 20 years ago, technological advances to make PCs more useful have continued at an amazing rate. Microprocessors that control PCs have become faster and faster, with operational speeds eclipsing the gigahertz (one billion operations per second) and continuing well beyond.

Productivity has also increased tremendously because of the explosion in development of software applications. In the early days of the PC, people who could write their own programs were practically the only ones who could make productive use of their computers. Today, there are thousands and thousands of software applications ranging from games to word processors and from voice recognition to web browsers.

5

In addition to improvements in PC hardware and software generally, the technology for making computers more useful by allowing users to connect PCs together and share resources between them has also seen rapid growth in recent years. This technology is generally referred to as “networking.” In a networked computing environment, PCs belonging to many users are connected together so that they may communicate with each other. In this way, users can share access to each other’s files and other resources, such as printers. Networked computing also allows users to share internet connections, resulting in significant cost savings. Networked computing has revolutionized the way in which business is conducted across the world.

10

15

20

Not surprisingly, the evolution of networked computing has presented technologists with some challenging obstacles along the way. One obstacle is connecting computers that use different operating systems (“OSes”) and making them communicate efficiently with each other. Each different OS (or even variations of the same OS from the same company) has its own idiosyncrasies of operation and configuration. The interconnection of computers running different OSes presents significant ongoing issues that make day-to-day management of a computer network challenging.

5

Another significant challenge presented by the evolution of computer networking is the sheer scope of modern computer networks. At one end of the spectrum, a small business or home network may include a few client computers connected to a common server, which may provide a shared printer and/or a shared internet connection. On the other end of the spectrum, a global company's network environment may require interconnection of hundreds or even thousands of computers across large buildings, a campus environment or even between groups of computers in different cities and countries. Such a configuration would typically include a large number of servers, each connected to numerous client computers.

10

15

Further, the arrangements of servers and clients in a larger network environment could be connected in any of an infinite number of topologies that may include local area networks (“LANs”), wide area networks (“WANs”) and municipal area networks (“MANs”). In these larger networks, a problem with any one server computer (for example, a failed hard drive, failed network interface card or OS lock-up to name just a few) has the potential to interrupt the work of a large number of workers who depend on network resources to get their jobs done efficiently. Needless to say, companies devote a lot of time and effort to keeping their networks operating trouble-free to maximize productivity.

20

An important aspect of efficiently managing a large computer network is to maximize the amount of analysis and repair that can be performed remotely (for example, from a centralized administration site). Tools that facilitate remotely analyzing and servicing server problems help to control network management costs by reducing the number of network management personnel required to maintain a network in good working order. Remote server management also makes

network management more efficient by reducing the delay and expense of analyzing and repairing network problems. Using remote management tools, a member of the network management team may identify problems and, in some cases, solve those problems without the delay and expense that accompanies an on-site service call to a distant location.

5

The growth of modern computer systems and the need to communicate remotely with servers on such a vast geographic scale has, unfortunately, resulted in some unintended negative consequences. Large, geographically diversified computer networks are susceptible to attacks by hackers who desire to gain access to information stored on those computer networks. Such 10 hackers are sometimes motivated by nothing more than the pride they feel when they accomplish a difficult and technically challenging goal in the face of significant efforts to repel their efforts. Sadly, however, many hackers seek to compromise network security for material gain.

10

The data stored on modern computer networks ranges from military and corporate secrets 15 to personal information, including social security numbers, bank records or information about credit card accounts. This information is obviously very valuable. If a conniving hacker successfully breaks into a network environment where such data is stored, the hacker may use the data directly or sell it to others who would not hesitate to use the information to the disadvantage of its rightful owner. The theft of supposedly secure data could subject the company that stored 20 the data to falling customer confidence resulting in declining business opportunities. Failure to adequately protect the confidential information of others can even give rise to legal liability in some cases.

The complexity of modern computer networks and the equipment that comprises those networks provides hackers with many, many avenues to access secured data. Hackers frequently seek to take advantage of operational quirks of computer hardware and software (such as browser software) that the designers of that hardware or software did not intend or foresee. Even aspects 5 of computer equipment that are intended to provide new features and additional security give imaginative hackers opportunities to compromise network security and steal confidential data.

Remote server management tools that communicate with network servers are a favorite 10 target of hackers' attention. This is true because remote server management tools often have the capability to give a user broad access rights to network servers. Server management requires broad network access rights of the type typically possessed by network administrators who have access to even the most sensitive data stored on a server.

A common practice employed by hackers is to obtain access to a network and eavesdrop 15 or "snoop" communications until they identify a communication that is likely to be a network management communication. If a knowledgeable hacker is able to compromise a management communication with a server, the hacker will undoubtedly have a much greater chance to gain access to sensitive data stored on that server.

20 The security problems previously discussed are magnified in a remote server management setting, especially for remote communications that are established using publicly available communication paths such as the Internet. The public nature of the Internet makes it a prime

target for hackers as they are able to snoop huge volumes of communications in their efforts to locate a vulnerable server.

Of course, the broad public availability of the Internet, which makes it of such interest to
5 hackers, is also of significant appeal to the designers of remote server management tools. A primary goal of remote server management professionals is to create tools to facilitate remote management in the broadest range of contexts possible. A tool which is able to provide remote management capabilities over the Internet is directly in keeping with that desire.

10 To facilitate the goal of providing remote server management across publicly accessible data paths such as the Internet while at the same time inhibiting the ability of hackers to intercept such communications, a remote network management tool that is able to establish a strongly encrypted communication path with a managed server across the publicly available communication path is highly desirable.

15

BRIEF DESCRIPTION OF THE DRAWINGS

The foregoing and other advantages of the invention will become apparent upon reading the following detailed description and upon reference to the drawings in which:

20 FIG. 1 is a block diagram illustrating an exemplary computer network system in which a
remote server management controller of the present invention may be practiced;

FIG. 2 is a functional block diagram of one embodiment of a remote server management controller constructed according to the present invention;

5 FIG. 3 is a functional block diagram showing an exemplary embodiment of hardware and software elements useful to establish a secure connection with a managed server across a publicly accessible data path such as the Internet.

10 FIG. 4 is a process flow diagram showing an exemplary process for the establishment of a secure connection with a managed server across a publicly accessible data path such as the Internet.

DESCRIPTION OF SPECIFIC EMBODIMENTS

One or more specific embodiments of the present invention will be described below. In an effort to provide a concise description of these embodiments, not all features of an actual 15 implementation are described in the specification. It should be appreciated that in the development of any such actual implementation, as in any engineering or design project, numerous implementation-specific decisions must be made to achieve the developers' specific goals, such as compliance with system-related and business-related constraints, which may vary from one implementation to another. Moreover, it should be appreciated that such a 20 development effort might be complex and time consuming, but would nevertheless be a routine undertaking of design, fabrication, and manufacture for those of ordinary skill having the benefit of this disclosure.

Turning now to the drawings and referring initially to FIG. 1, a block diagram of an exemplary computer network system in which the present invention may be practiced is illustrated and designated using a reference numeral 10. The computer network 10 is intended to illustrate a typical modern computer network configuration with all its complexities and 5 flexibility. A managed server 20 is connected to a plurality of client computers 22, 24 and 26. For purposes of explaining the present invention clearly, only one server on the network 10 has been designated as a “managed server.” In practice, those of skill in the art will appreciate that the any or all of the servers in the network 10 could simultaneously include hardware and software devised according to the invention, making those servers “managed servers” as well.

10

15

The managed server 20 may be connected to as many as n different client computers. The magnitude of n is a function of the computing power of the managed server 20. If the managed server has large computing power (for example, faster processor(s) and/or more system memory) relative to other servers on the network, it will be able to effectively serve a relatively large number of client computers.

20

The managed server 20 is connected via a typical network infrastructure 30, which may consist of any combination of hubs, switches, routers and the like. While the network infrastructure 30 is illustrated as being either a local area network (“LAN”) or wide area network (“WAN”), those skilled in the art will appreciate that the network infrastructure 30 may assume other forms such as a municipal area network (“MAN”) or even the Internet.

The network infrastructure 30 connects the managed server 20 to server 40, which is representative of any other server in the network environment of managed server 20. The server 40 may be connected to a plurality of client computers 42, 44 and 46. The server 40 is additionally connected to server 50, which is in turn connected to client computers 52 and 54.

5 The number of client computers connected to the servers 40 and 50 is dependent only on the computing power of the servers 40 and 50, respectively.

10 The server 40 is additionally connected to the Internet 60, which is in turn connected to a server 70. Server 70 is connected to a plurality of client computers 72, 74 and 76. As with the other servers shown in FIG. 1, server 70 may be connected to as many client computers as its computing power will allow.

15 Those skilled in the art will appreciate that neither the exact topology of the network illustrated in FIG. 1 nor the protocol of network communication (for example, Ethernet or any number of other common protocols) is a crucial aspect of the present invention. Moreover, the network topology shown in FIG. 1 is hypothetical and is shown only to explain the present invention with greater clarity by giving an example of a network environment in which the present invention is useful.

20 As explained in detail below, the present invention is illustrated as being deployed in a remote server management controller by way of example. The exemplary remote server management controller may be hosted in the managed server 20. The exemplary remote server

management controller may be accessed via a remote console application program (or browser program) running on any of the client computers shown in FIG. 1.

For purposes of explaining the invention clearly, the remainder of this description is written with the assumption that the server 70 and clients 72, 74 and 76 are located in a remote administration site. The further assumption is made that a user (network administrator) located at the network administration site desires to use a remote console program running on one of the clients 72, 74 or 76 to establish a remote server management communication with the managed server 20. To accomplish this connection, the communication path between server 70 and managed server 20 must traverse the Internet 60, which is publicly accessible and prone to eavesdropping and snooping by hackers. After traversing the Internet 60, the communication path between the server 70 and the managed server 20 traverses servers 40, 50 and network infrastructure 30 in a typical manner before terminating at the managed server 20.

FIG. 2 shows a functional block diagram of one exemplary embodiment of a remote server management controller 200 constructed according to the present invention. The remote server management controller 200 may be implemented in a single application specific integrated circuit (“ASIC”). Alternatively, the remote server management controller 200 may be implemented in a plurality of integrated circuits or discrete components. Those skilled in the art will appreciate that implementation details such as deciding which functional aspects of remote server management controller 200 are implemented in a single ASIC or different ASICs are matters of design choice and not crucial aspects of the present invention.

The remote server management controller 200 may be implemented so that it is powered and capable of operation whether or not the managed server 20 (FIG. 1) is powered up (turned on) or online. Powering the remote server management controller 200 regardless of whether the host managed server is turned on allows the remote server management controller 200 to monitor, analyze and potentially intervene to correct the widest possible range of system problems that may befall the managed server 20 (FIG. 1).

The logic of the remote server management controller 200 is broken down into three main functional blocks. The first of these three functional blocks is an embedded I/O controller 300, which is essentially an independent computer system that is integrated within the managed server 20 (FIG. 1). The second and third functional blocks of the remote server management controller 200 are a slave instrumentation module 400 and a remote console redirection module 500. As described below, the embedded I/O controller 300 monitors and controls a wide range of conditions in the managed server 20 via the slave instrumentation module 400 and the remote console redirection module 500.

The embedded I/O controller 300 comprises an Input/Output processor (“IOP”) 302, which provides general control and functions as a management processor for the remote server management controller 200. The IOP 302 may be implemented as a 32-bit RISC processor, but other processor implementations may be employed as well. The IOP 302 is operatively coupled to a timer module 304 and an interrupt controller 306 via a peripheral bus 308.

In one exemplary embodiment of the invention, a memory controller 309 is operatively coupled to the internal local bus 310. The memory controller 309 is, in turn, operatively coupled to dedicated memory via a memory interface 311. The dedicated memory may be battery-backed SRAM, SDRAM, ROM, NVRAM or any other appropriate type of memory.

5

The IOP 302 (located in the embedded I/O controller 300) is operatively coupled to the other functional modules (and many sub-modules) of the remote server management controller 200 via an internal local bus 310. Those of ordinary skill in the field will appreciate that the internal local bus 310 exists to allow communication between and among the logical components of the remote server management controller 200. The implementation details of the internal local bus 310 are a matter of design choice and not a crucial aspect of the present invention.

10
15
20
25
30
35
40
45
50
55
60
65
70
75
80
85
90
95

An address translation and bridging (“ATB”) unit 312 is operatively coupled to the internal local bus 310 and to a PCI bus 314. PCI bus 314 is integral within and operatively coupled with the managed server 20 (FIG. 1). Preferably, the PCI bus 314, which serves as the main communication interface between the managed server 20 (FIG. 1) and the remote server management controller 200, may be configured as a 32-bit, 33 MHz PCI master/slave interface. In a typical system implementation, the remote server management controller 200 resides on the “compatibility” segment of PCI bus 314, but the PCI bus segment on which the remote server management controller is disposed is not a crucial aspect of the invention. The ATB unit 312 is constructed to allow the remote server management controller 200 to decode bus cycles on the PCI bus 314 and to communicate over the PCI bus 314 by initiating PCI bus cycles.

The embedded I/O controller 300 provides a plurality of communication interfaces that can be employed to establish communication sessions with the remote server management controller 200. One such communication interface is a UART interface module 316, which is operatively coupled to internal local bus 310. The exemplary UART interface module 316 comprises two standard 16550 UARTs, each of which may provide a separate serial communication interface between the remote server management controller 200 and the external world. Both UARTs are mapped into the address space of the IOP 302 and can be accessed via PCI bus 314 or by the IOP 302. Either UART may be implemented so that it can be reset through a control register in the address space of the IOP 302.

Outputs from the UART interface module 316 are typically routed to transceivers (not shown), where they may be converted into a wide variety of serial interface types. Examples of the types of serial interfaces that may be provided by the UART interface module 316 are a standard RS-232 interface 318 or an interface that complies with the Intelligent Chassis Management Bus (“ICMB”) specification promulgated by Intel Corporation (ICMB interface 320). Those of ordinary skill in the field will appreciate that the RS-232 interface 318 may be used to connect to a wide range of industry standard modems, terminal servers and the like.

Embedded I/O controller 300 further comprises an Ethernet interface 322, which is operatively coupled to the internal local bus 310. Ethernet interface 322 provides the main external communication interface between the remote server management controller 200 and the outside world. In the exemplary embodiment shown in FIG. 2, the integrated portion of the Ethernet interface 322 includes a MAC (Media Access Controller), inbound and outbound

FIFOs, and a DMA engine to automatically transfer packets to and from memory. The Ethernet interface 322 requires a connection via interface 328 to an external PHY (not shown) and typical magnetic coupling to couple the PHY to the wire that serves as the transmission media.

5 Those skilled in the art will appreciate that a user may connect remotely to the remote server management controller 200 via the Ethernet interface 322. Such a connection may be made, for example, using a remote console application running on a client computer anywhere on the network that includes managed server 20 (FIG. 1). Moreover, the exemplary embodiment is capable of establishing secure communications between servers on the same network even when those servers are not connected to or by a publicly available data path such as the Internet. For purposes of clearly explaining the present invention, however, this description assumes that a remote user desires to establish a management communication session with the managed server 20 across the Internet 60 (FIG. 1). Those of ordinary skill in the field will appreciate that the Ethernet interface 322 may be configured to transmit and receive data according to the well-known TCP/IP protocol to allow the remote server management controller to be accessed via the Internet 60 (FIG. 1).

10

15

20 Embedded I/O controller 300 further comprises a USB interface 326, which is operatively coupled to the internal local bus 310. The USB interface 326 is connected to a USB host controller (not shown) via a USB host controller interface 328. The USB interface 326 is connected to one port of a USB host controller, which is typically located in a south bridge portion of the chipset of the managed server 20 (FIG. 1). When implemented in this way, the IOP 302 of the remote server management controller 200 may establish “virtual USB

“peripherals” that will be seen and recognized by any USB-aware OS or properly adapted system BIOS. These virtual peripherals may be presented to any USB-aware OS to allow communication with the OS in a common, OS-independent manner.

5 The next major functional block of the remote server management controller 200 is the slave instrumentation module 400. The primary purpose of the slave instrumentation module 400 is to provide the hardware infrastructure to implement control and monitoring functions in the managed server 20 (FIG. 1) as dictated by the IOP 302 in conjunction with dedicated application software such as remote console management software running on a client computer.

10 The slave instrumentation module 400 comprises an automatic server recovery (“ASR”) controller 402, which operates to automatically respond to catastrophic failures of the managed server 20 (FIG. 1). The ASR controller 402 is operatively coupled to the internal local bus 310. The ASR controller 402 continually monitors whether the OS of the managed server 20 (FIG. 1) is operational by controlling a dead-man timer that requires periodic servicing by the OS. If the OS of the managed server 20 (FIG. 1) does not service the dead-man timer within a predetermined time, the ASR controller 402 resets the processor of the managed server 20 (FIG. 1) causing the managed server 20 (FIG. 1) to reboot.

15 A general purpose input/output module (“GPIO”) 405 is provided in the exemplary embodiment of the slave instrumentation module 400. The GPIO provides a versatile communication interface that may be used for a wide variety of purposes.

The slave instrumentation module 400 also comprises a JTAG master 404. The JTAG master 404 is operatively coupled to the internal local bus 310. The JTAG master 404 comprises a standard JTAG interface 406, which is operatively coupled to a corresponding standard JTAG interface (not shown) on the motherboard of the managed server 20 (FIG. 1). Through the JTAG master 404, the remote server management controller 200 can perform a wide range of control functions on the managed server 20 (FIG. 1). These functions include updating or repairing the BIOS of the managed server 20 by reprogramming the non-volatile memory where the BIOS resides.

The slave instrumentation module 400 further comprises an I²C master 408, which is operatively coupled with the internal local bus 310. The I²C master 408 has the capability of controlling a plurality of independent I²C serial channels 410. For purposes of example only, four (4) separate I²C channels are shown in FIG. 2. The I²C master 408 comprises a separate I²C engine for controlling each separate I²C channel.

The slave instrumentation module 400 additionally comprises a block of system support logic 412. The system support logic 412 is operatively coupled to the internal local bus 310. The system support logic 412 provides a variety of housekeeping functions for the remote server management controller 200. Examples of these functions include providing the EISA bus ID, flash ROM support, ECC support, hot spare boot support, system post monitor support, floppy write protect, SMI base security measures, open hood detection and the like.

The third and final major functional block of the remote server management controller 200 is the remote console redirection module 500, which comprises a video encoder 502 and integrated remote console (“IRC”) registers 504. The IRC registers 504 receive raw data snooped from the PCI bus 314. Under control of the IOP 302, some of the IRC registers 504 may 5 function as a virtual communication device (“VCD”) that may be used to intercept UART communications. Data intercepted through the VCD may be altered by the IOP and/or redirected to other outputs of the remote server management controller 200. For example, data intercepted by the VCD may be redirected to a remote user via the Ethernet interface 322.

10 FIG. 3 is a functional block diagram showing an exemplary embodiment of hardware and software elements useful to establish a secure connection between the client computer 72 (FIG. 1) and the managed server 20 (FIG. 1) across a publicly accessible data path such as the Internet 60 (FIG. 1). In FIG. 3 the client computer 72 (FIG. 1) is connected directly to the Internet 60, which is, in turn, connected to managed server 20 (FIG. 1). For purposes of clarity, the 15 intervening connections through the servers 40 and 50 (FIG. 1) and the network interface 30 (FIG. 1) are not shown in FIG. 3.

20 The client computer 72 is shown in FIG. 3 (as opposed to any of the other clients illustrated in FIG. 1) for illustrative and explanatory purposes only. Those of ordinary skill in the art will appreciate that any client computer connected to the network environment of the managed server 20 (FIG. 1) or with access to the Internet 60 (FIG. 1) may be used to establish a secure management communication session with managed server 20 (FIG. 1).

In the exemplary embodiment, the remote server management controller 200 provides a web server 600. The web server 600 may be a functional block of the firmware stack that is executing on the IOP 302 (FIG. 2). The web server 600 provides the functionality to (1) communicate on the Internet 60 and (2) serve up information and data (for example, web pages or Java applets) to be displayed to a user in response to requests made by the user via a browser application 604. The browser application 604 may be any of a wide variety of browser applications that are generally available (for example, Netscape by AOL or Internet Explorer by Microsoft).

In the exemplary embodiment shown in FIG. 3, a remote console server 602 is implemented by the remote console redirection module 500 (FIG. 2) in conjunction with the slave implementation module 400. The remote console server 602 has the capability of providing a user with management and condition (health) data gathered by the slave instrumentation module 400. The web server 600 and the remote console server 602 may exchange information with each other over data path 601, which may comprise the internal local bus 310 (FIG. 2).

As described above, the remote server management controller 200 may obtain textual and graphical video images and reset sequence data from managed server 20 (FIG. 1). In addition, keyboard input (keystroke) data and mouse position information may also be obtained by the remote server management controller 200.

5

The remote console server 602 may operate to transmit gathered textual video data, graphical video data, keystroke information and/or mouse movement information to a user, who is accessing the remote server management controller 200 using a remote console applet 606. The user may also communicate control information to the remote console server 602 via the remote console applet 606.

10

The remote console applet 606 is an application program that is invoked by a user to communicate with the remote console server 602. In the illustrative situation shown in FIG. 3, the user communicates with the remote console server 602 located in the remote server management controller 200 by running remote console applet 606 on client computer 72 (FIG. 1). Examples of programs that may function as remote console applets are Carbon Copy, VNC (see <http://www.uk.research.att.com/vnc/>) and PC-Anywhere. The remote console applet 606 has the ability to communicate with the browser 604 via a communication path 605. The communication path 605 may comprise the message passing capability of the OS of the client computer 72.

15

20

As shown in FIG. 3, the browser application 604 may establish a communication session with web server 600 across the Internet 60. For purposes of illustration, the communication channel between the browser application 604 and the web server 600 is designated with reference numeral 608.

The browser application 604 typically supports a secure communication protocol called SSL, which stands for Secure Sockets Layer. The SSL protocol is well-known in the art and is

commonly used to facilitate secure transactions over the Internet 60. While SSL is used to encrypt communications between the browser 604 and the web server 600 in the exemplary embodiment, those of ordinary skill in the art will appreciate that other secure protocols may be used instead of SSL. The choice of secure protocol is not a crucial feature of the present invention.

5

The SSL protocol supports the establishment of an encrypted communication path between the browser 604 and a server that is SSL-compatible. In the exemplary embodiment shown in FIG. 3, the web server 600 is SSL-compatible.

10

The first step in establishing a secure communication between the browser 604 and the web server 600 is the establishment of a typical non-secure communication link using a protocol such as TCP/IP. After a non-secure connection is established between the browser 604 and an SSL server (such as the web server 604), the browser 604 and the web server 600 negotiate to initiate a secure SSL communication link (a “socket”). The SSL protocol supports a large number of encryption algorithms and other security functions such as digital certificate generation. As part of the initial negotiation, the browser 604 and the web server 600 agree to use a common encryption algorithm and other parameters necessary to facilitate secure communication.

15

20

After the browser 604 and the web server 600 agree on the encryption algorithm and other necessary parameters, the browser 604 requests the public key of the web server 600. The web server 600 responds and may send the public key in the form of a digital certificate.

Next, the browser 604 sends the web server 600 a secret (for example, a random number) that has been encrypted with the server's public key. The server decrypts the secret using its private key (typically, a mathematical inverse of its public key). The browser 604 and web server 600 each use the secret sent by the browser 604 to generate a shared secret key. The browser 604 authenticates that the web server 600 has generated the same secret key. After authentication, the secret key is used to encrypt subsequent communications between the browser 604 and the web server 600.

100-200-300-400-500-600-700-800-900-1000

15

The browser 604 and web server 600 each may in fact generate a plurality of shared secret keys for different purposes (for example, encryption, message integrity and/or cipher initialization). For purposes of illustration, the example in this discussion refers only to the generation of a single shared secret key. Also, the number of secret keys generated by the browser and/or the server is not a crucial aspect of the invention. Once the socket is established between the browser 604 and the web server 600, information transmitted across the socket is practically indecipherable to eavesdroppers such as hackers because it is computationally too expensive to decipher the packets without prior knowledge of the keys being used.

20

In the exemplary embodiment of the invention shown in FIG. 3, the remote console applet 606 may separately establish an independent communication path with a remote server console 602. This communication path may be used by a user to retrieve management data from and send control data to the remote server management controller 200. For purposes of illustration, the communication channel between the remote console applet 604 and the remote console server 602 is designated with reference numeral 610.

When the remote console applet 606 first establishes the communication link 610, the communication link 610 is “in the clear,” which means that communication thereon is not encrypted. It is desirable to make the communication link 610 secure because the link 610 is used to transmit management and control data (for example, textual video data, graphical video data, keystroke data, mouse movement data and command and control data) of a most sensitive nature during remote management communication sessions with the remote server management controller 200. The management and control data transmitted across communication link 610 would, if intercepted, improve a hacker’s chances of gaining access to secure data stored on the managed server 20 (FIG. 1).

10

15

The communication link 610 could be secured in the manner described above with respect to the establishment of secure communication using the SSL protocol. To secure the communication link 610 in that manner would, however, require the remote console applet 606 and the remote console server 602 to have access to executable code containing the full suite of SSL functionality, which includes multiple versions of SSL executable code, digital certificate management, key exchange capability and dozens of encryption algorithms. Most of that functionality is not required to secure the communication link 610.

20

The remote console server 602 has limited memory resources because it is deployed as part of remote server management controller 200. Minimizing the size of executable code that must be stored in the memory of the remote server management controller 200 is a desirable goal for purposes of making the remote server management controller 200 as self-contained as possible. Transfer time of the code across slow data links is also shortened by making the code

that must be transferred as small as possible. To help facilitate the goal of conserving memory space, the executable code implemented in the exemplary remote console server 602 comprises an encryption algorithm and not the entire suite of executable code needed for full SSL compatibility. In the exemplary embodiment, the executable code available to the remote console server 602 may be as little as a single encryption algorithm. That encryption may be the RC4 algorithm, but any other encryption algorithm may be used. The exact encryption algorithm used is not a crucial aspect of the invention.

The approach of transferring as little as a single encryption algorithm from the remote

10 console server 602 eliminates the need to rely on the native capabilities of JAVA functionality in many browser applications. JAVA, as implemented in many web browsers, supports web page browsing but not full duplex data exchange.

In the exemplary embodiment shown in FIG. 3, the executable code available to the

15 remote console server 602 does not necessarily include functionality that supports secure key exchange. If key exchange functionality is not available to the remote console server 602, some other mechanism for secure key exchange must be employed in order to secure the communication link 610 between the remote console applet 606 and the remote console server 602.

20

An exemplary process for securing the communication link 610 is shown in FIG. 4. The process illustrated in FIG. 4 (generally referred to by reference numeral 700) allows secure key

exchange between the remote console applet 606 and the remote console server 602 even if neither has access to software that provides secure key exchange functionality.

5 A user, who desires to establish a management communication session with the remote server management controller 200 (FIG. 3), launches the browser 604 from client computer 72 (FIG. 3). The user points the browser 604 at the URL of the remote management server controller 200. As shown at 701, the browser 604 establishes an SSL communication path 608 with the web server 600 via the Internet 60. The establishment of the communication path 608 may be done as described above with reference to FIG. 3.

10 At 702, the user engages the remote console applet 606, which establishes a clear (not encrypted) communication path 610 with the remote console server 602. As shown at 704, the remote console applet 606 communicates with the browser 604 over the communication link 605 and requests the browser 604 to obtain a secret key from the web server 600.

15 While the remote console applet 606 requests a secret key in the exemplary embodiment shown in FIG. 3 and FIG. 4, the request to the web server 600 could alternatively be for a secret (such as a random number) that would allow generation of a secret key as opposed to the secret key itself. In a further alternative, the request for a secret or secret key could be made by the 20 remote console server 602 instead of the browser.

The web server 600 responds (at shown at 706) by providing the requested secret key to the browser 604 via encrypted communication link 608. The secret key is also passed from the web server 600 to the remote console server 602 via the communication link 601 (706).

5 As shown at 707, the browser 604 passes the secret key on to the remote console applet 606 via communication link 605. At 708, the remote console applet 606 and the remote console server 602 use the shared secret key that both have obtained from the web server 600 to begin encrypted communication across the communication link 610. Thus, communication link 610 is upgraded from “in the clear” to secure. The user may proceed to obtain sensitive management data and transmit control information to the remote server management controller 200.

10 Those of ordinary skill in the art will observe that the secret key requested by the remote console applet 606 from the browser 600 is transmitted only over communication paths that are either encrypted (for example, communication path 608) or internal to a single computer (for example, the communication paths 601 and 605). In this manner, hackers are effectively denied the opportunity to obtain the secret key and use it to decrypt sensitive data transferred between the remote console applet and the remote console server 602 during a management communication session.

15 20 To further thwart the effort of hackers to decrypt the information transmitted between the remote console applet 606 and the remote console server 602, the shared secret key requested by the remote console applet may be given a relatively short effective period (for example, a few minutes). The useful life of the secret key may be limited through a variety of methods such as

through the use of timestamps. If additional time is needed for the management communication session, a new secret key could be requested prior to the expiration of the previous secret key. In this manner, the management communication session could continue uninterrupted.

5 As an alternative to requesting a new secret key prior to expiration of the old one, the remote console server 602 and the remote console applet 606 could either generate a plurality of secret keys upon receipt of the original secret key (or secret). The remote console server 602 and the remote console applet 606 could begin using the additional keys at agreed-upon times.

10 As an additional alternative, the remote console server 602 and the remote console applet 606 could incorporate functionality that would allow them to periodically update the secret key using an agreed-upon one-way mathematical transform (a hash). Examples of one-way transforms that could be used are the MD-5 algorithm or the SHA-1 algorithm. This key generation approach exploits the fact that the original exchange of a pre-master-secret (“PMS”) 15 (the originally requested key or secret) is transferred over a link that is encrypted between the browser 604 and the web server 600, so it is never in the clear. After receipt of the PMS, the applet creates a key by hashing the PMS with a known key value such as all zeroes. That key is used for a period of time that may be previously agreed upon. When it is time to generate a new key, the same computation is performed again: the then-existing key is hashed with the PMS and 20 the result is the new key. This process may be expressed as follows:

KEY \square PMS \square KEY

In the above expression, the \square symbol represents the one-way hash function. Because it is impossible to recover the input to the one-way-hash function from its output, it is very difficult or impossible to recover the pre-master-secret.

5

While the invention may be susceptible to various modifications and alternative forms, specific embodiments have been shown by way of example in the drawings and will be described in detail herein. However, it should be understood that the invention is not intended to be limited to the particular forms disclosed. Rather, the invention is to cover all modifications, equivalents and alternatives falling within the spirit and scope of the invention as defined by the following appended claims.

10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
259
260
261
262
263
264
265
266
267
268
269
269
270
271
272
273
274
275
276
277
278
279
279
280
281
282
283
284
285
286
287
288
289
289
290
291
292
293
294
295
296
297
298
299
299
300
301
302
303
304
305
306
307
308
309
309
310
311
312
313
314
315
316
317
318
319
319
320
321
322
323
324
325
326
327
328
329
329
330
331
332
333
334
335
336
337
338
339
339
340
341
342
343
344
345
346
347
348
349
349
350
351
352
353
354
355
356
357
358
359
359
360
361
362
363
364
365
366
367
368
369
369
370
371
372
373
374
375
376
377
378
379
379
380
381
382
383
384
385
386
387
388
389
389
390
391
392
393
394
395
396
397
398
399
399
400
401
402
403
404
405
406
407
408
409
409
410
411
412
413
414
415
416
417
418
419
419
420
421
422
423
424
425
426
427
428
429
429
430
431
432
433
434
435
436
437
438
439
439
440
441
442
443
444
445
446
447
448
449
449
450
451
452
453
454
455
456
457
458
459
459
460
461
462
463
464
465
466
467
468
469
469
470
471
472
473
474
475
476
477
478
479
479
480
481
482
483
484
485
486
487
488
489
489
490
491
492
493
494
495
496
497
498
499
499
500
501
502
503
504
505
506
507
508
509
509
510
511
512
513
514
515
516
517
518
519
519
520
521
522
523
524
525
526
527
528
529
529
530
531
532
533
534
535
536
537
538
539
539
540
541
542
543
544
545
546
547
548
549
549
550
551
552
553
554
555
556
557
558
559
559
560
561
562
563
564
565
566
567
568
569
569
570
571
572
573
574
575
576
577
578
579
579
580
581
582
583
584
585
586
587
588
589
589
590
591
592
593
594
595
596
597
598
599
599
600
601
602
603
604
605
606
607
608
609
609
610
611
612
613
614
615
616
617
618
619
619
620
621
622
623
624
625
626
627
628
629
629
630
631
632
633
634
635
636
637
638
639
639
640
641
642
643
644
645
646
647
648
649
649
650
651
652
653
654
655
656
657
658
659
659
660
661
662
663
664
665
666
667
668
669
669
670
671
672
673
674
675
676
677
678
679
679
680
681
682
683
684
685
686
687
688
689
689
690
691
692
693
694
695
696
697
698
698
699
699
700
701
702
703
704
705
706
707
708
709
709
710
711
712
713
714
715
716
717
718
719
719
720
721
722
723
724
725
726
727
728
729
729
730
731
732
733
734
735
736
737
738
739
739
740
741
742
743
744
745
746
747
748
749
749
750
751
752
753
754
755
756
757
758
759
759
760
761
762
763
764
765
766
767
768
769
769
770
771
772
773
774
775
776
777
778
779
779
780
781
782
783
784
785
786
787
788
789
789
790
791
792
793
794
795
796
797
798
798
799
799
800
801
802
803
804
805
806
807
808
809
809
810
811
812
813
814
815
816
817
818
819
819
820
821
822
823
824
825
826
827
828
829
829
830
831
832
833
834
835
836
837
838
839
839
840
841
842
843
844
845
846
847
848
849
849
850
851
852
853
854
855
856
857
858
859
859
860
861
862
863
864
865
866
867
868
869
869
870
871
872
873
874
875
876
877
878
879
879
880
881
882
883
884
885
886
887
888
889
889
890
891
892
893
894
895
896
897
898
898
899
899
900
901
902
903
904
905
906
907
908
909
909
910
911
912
913
914
915
916
917
918
919
919
920
921
922
923
924
925
926
927
928
929
929
930
931
932
933
934
935
936
937
938
939
939
940
941
942
943
944
945
946
947
948
949
949
950
951
952
953
954
955
956
957
958
959
959
960
961
962
963
964
965
966
967
968
969
969
970
971
972
973
974
975
976
977
978
979
979
980
981
982
983
984
985
986
987
988
988
989
989
990
991
992
993
994
995
996
997
998
998
999
999
1000
1001
1002
1003
1004
1005
1006
1007
1008
1009
1009
1010
1011
1012
1013
1014
1015
1016
1017
1018
1019
1019
1020
1021
1022
1023
1024
1025
1026
1027
1028
1029
1029
1030
1031
1032
1033
1034
1035
1036
1037
1038
1039
1039
1040
1041
1042
1043
1044
1045
1046
1047
1048
1049
1049
1050
1051
1052
1053
1054
1055
1056
1057
1058
1059
1059
1060
1061
1062
1063
1064
1065
1066
1067
1068
1069
1069
1070
1071
1072
1073
1074
1075
1076
1077
1078
1079
1079
1080
1081
1082
1083
1084
1085
1086
1087
1088
1088
1089
1089
1090
1091
1092
1093
1094
1095
1096
1097
1098
1098
1099
1099
1100
1101
1102
1103
1104
1105
1106
1107
1108
1109
1109
1110
1111
1112
1113
1114
1115
1116
1117
1118
1119
1119
1120
1121
1122
1123
1124
1125
1126
1127
1128
1129
1129
1130
1131
1132
1133
1134
1135
1136
1137
1138
1139
1139
1140
1141
1142
1143
1144
1145
1146
1147
1148
1149
1149
1150
1151
1152
1153
1154
1155
1156
1157
1158
1159
1159
1160
1161
1162
1163
1164
1165
1166
1167
1168
1169
1169
1170
1171
1172
1173
1174
1175
1176
1177
1178
1179
1179
1180
1181
1182
1183
1184
1185
1186
1187
1188
1188
1189
1189
1190
1191
1192
1193
1194
1195
1196
1197
1198
1198
1199
1199
1200
1201
1202
1203
1204
1205
1206
1207
1208
1209
1209
1210
1211
1212
1213
1214
1215
1216
1217
1218
1219
1219
1220
1221
1222
1223
1224
1225
1226
1227
1228
1229
1229
1230
1231
1232
1233
1234
1235
1236
1237
1238
1239
1239
1240
1241
1242
1243
1244
1245
1246
1247
1248
1249
1249
1250
1251
1252
1253
1254
1255
1256
1257
1258
1259
1259
1260
1261
1262
1263
1264
1265
1266
1267
1268
1269
1269
1270
1271
1272
1273
1274
1275
1276
1277
1278
1279
1279
1280
1281
1282
1283
1284
1285
1286
1287
1288
1288
1289
1289
1290
1291
1292
1293
1294
1295
1296
1297
1298
1298
1299
1299
1300
1301
1302
1303
1304
1305
1306
1307
1308
1309
1309
1310
1311
1312
1313
1314
1315
1316
1317
1318
1319
1319
1320
1321
1322
1323
1324
1325
1326
1327
1328
1329
1329
1330
1331
1332
1333
1334
1335
1336
1337
1338
1339
1339
1340
1341
1342
1343
1344
1345
1346
1347
1348
1349
1349
1350
1351
1352
1353
1354
1355
1356
1357
1358
1359
1359
1360
1361
1362
1363
1364
1365
1366
1367
1368
1369
1369
1370
1371
1372
1373
1374
1375
1376
1377
1378
1379
1379
1380
1381
1382
1383
1384
1385
1386
1387
1388
1388
1389
1389
1390
1391
1392
1393
1394
1395
1396
1397
1398
1398
1399
1399
1400
1401
1402
1403
1404
1405
1406
1407
1408
1409
1409
1410
1411
1412
1413
1414
1415
1416
1417
1418
1419
1419
1420
1421
1422
1423
1424
1425
1426
1427
1428
1429
1429
1430
1431
1432
1433
1434
1435
1436
1437
1438
1439
1439
1440
1441
1442
1443
1444
1445
1446
1447
1448
1449
1449
1450
1451
1452
1453
1454
1455
1456
1457
1458
1459
1459
1460
1461
1462
1463
1464
1465
1466
1467
1468
1469
1469
1470
1471
1472
1473
1474
1475
1476
1477
1478
1479
1479
1480
1481
1482
1483
1484
1485
1486
1487
1488
1488
1489
1489
1490
1491
1492
1493
1494
1495
1496
1497
1498
1498
1499
1499
1500
1501
1502
1503
1504
1505
1506
1507
1508
1509
1509
1510
1511
1512
1513
1514
1515
1516
1517
1518
1519
1519
1520
1521
1522
1523
1524
1525
1526
1527
1528
1529
1529
1530
1531
1532
1533
1534
1535
1536
1537
1538
1539
1539
1540
1541
1542
1543
1544
1545
1546
1547
1548
1549
1549
1550
1551
1552
1553
1554
1555
1556
1557
1558
1559
1559
1560
1561
1562
1563
1564
1565
1566
1567
1568
1569
1569
1570
1571
1572
1573
1574
1575
1576
1577
1578
1579
1579
1580
1581
1582
1583
1584
1585
1586
1587
1588
1588
1589
1589
1590
1591
1592
1593
1594
1595
1596
1597
1598
1598
1599
1599
1600
1601
1602
1603
1604
1605
1606
1607
1608
1609
1609
1610
1611
1612
1613
1614
1615
1616
1617
1618
1619
1619
1620
1621
1622
1623
1624
1625
1626
1627
1628
1629
1629
1630
1631
1632
1633
1634
1635
1636
1637
1638
1639
1639
1640
1641
1642
1643
1644
1645
1646
1647
1648
1649
1649
1650
1651
1652
1653
1654
1655
1656
1657
1658
1659
1659
1660
1661
1662
1663
1664
1665
1666
1667
1668
1669
1669
1670
1671
1672
1673
1674
1675
1676
1677
1678
1679
1679
1680
1681
1682
1683
1684
1685
1686
1687
1688
1688
1689
1689
1690
1691
1692
1693
1694
1695
1696
1697
1698
1698
1699
1699
1700
1701
1702
1703
1704
1705
1706
1707
1708
1709
1709
1710
1711
1712
1713
1714
1715
1716
1717
1718
1719
1719
1720
1721
1722
1723
1724
1725
1726
1727
1728
1729
1729
1730
1731
1732
1733
1734
1735
1736
1737
1738
1739
1739
1740
1741
1742
1743
1744
1745
1746
1747
1748
1749
1749
1750
1751
1752
1753
1754
1755
1756
1757
1758
1759
1759
1760
1761
1762
1763
1764
1765
1766
1767
1768
1769
1769
1770
1771
1772
1773
1774
1775
1776
1777
1778
1779
1779
1780
1781
1782
1783
1784
1785
1786
1787
1788
1788
1789
1789
1790
1791
1792
1793
1794
1795
1796
1797
1798
1798
1799
1799
1800
1801
1802
1803
1804
1805
1806
1807
1808
1809
1809
1810
1811
1812
1813
1814
1815
1816
1817
1818
1819
1819
1820
1821
1822
1823
1824
1825
1826
1827
1828
1829
1829
1830
1831
1832
1833
1834
1835
1836
1837
1838
1839
1839
1840
1841
1842
1843
1844
1845
1846
1847
1848
1849
1849
1850
1851
1852
1853
1854
1855
1856
1857
1858
1859
1859
1860
1861
1862
1863
1864
1865
1866
1867
1868
186